REMARKS/ARGUMENTS

Applicants amended the Abstract to comply with the requirements the Examiner noted. Applicants amended the Specification to correct the informality noted by the Examiner.

1. Claims 1-3, 5-8, 10-13, and 15 are Patentable Over the Cited Art

The Examiner rejected claims 1-3, 5-8, 10-13, and 15 as obvious (35 U.S.C. §103(a)) over Boyer (U.S. Patent No. 5,774,692) in view of Cheng (U.S. Patent No. 5,963,933).

Independent claims 1, 6, and 11 recite concern translating a path expression in an object oriented query to a relational database outer join, said path expression comprising a navigation path through a relationship in a schema. The claims require: analyzing each path expression defined in each level of the object oriented query; identifying each path expression which can be a candidate for a translation to an outer join; ordering the path expression starting with path expression defined in a FROM clause, adding to the FROM clause path expression, each path expression identified as a candidate for a translation to an outer join, and making the ordered path expressions as input to a select operator for each level of the object oriented query; grouping the ordered path expressions sequentially based upon on a source-target dependency between ordered path expressions and based upon the identifications as a candidate for a translation to an outer join; creating a quantifier for each path expression, said quantifier comprising a variable representing a table in a relational database; replacing each grouped path expression with a corresponding quantifier and related table in a relational database; and completing a translation of the object oriented query to a relational query.

Applicants amended claims 1 and 6 to edit the preamble.

The Examiner cited col. 9, line 34 and 46-65 and col. 10, line 53 to col. 11, line 54 of Boyer as teaching the claim requirement of ordering the path expression starting with path expression defined in a FROM clause, adding to the FROM clause path expression, each path expression identified as a candidate for a translation to an outer join, and making the ordered path expressions as input to a select operator for each level of the object oriented query. (Office Action, pg. 3)

The cited col. 9 mentions a from statement “from Supplier s, (s.parts) p”, where quantifier “s” is declared on suppliers and quantifier “p” is declared on the parts that “s” supplies. A “quantifier” allows one to define an alias for a table. (Boyer, col. 1, lines 62-67)

The cited col. 9 further mentions that the query requests suppliers and the parts that they supply. If a supplier supplies no parts, the supplier will not appear in the result. A from statement "From outer Supplier s, (s.parts) p)" is a declaration of the collection of suppliers prefixed with the "outer" keyword. Since supplier is designated as "outer", and since "part" is dependent on "supplier", "part" is also "outer".

The cited col. 10 mentions that outer quantifiers can be implemented by a nested loop. Each quantifier in a query is implemented using an iterator or cursor that loops through a set of elements. Qualifiers over nested collections introduce a partial ordering among quantifiers.

Nowhere do the cited cols. 9 and 10 anywhere teach or suggest ordering path expressions starting with a path expression defined in a FROM clause and adding to the FROM clause path expressions, where each path expression identified for a translation to an outer join and making the ordered path expressions as input to a select operator. There is no mention or teaching of adding to a FROM statement path expressions identified as a candidate for a translation to an outer join. Instead, the cited cols. 9 and 10 discuss how qualifiers such as outer may apply to quantifiers in the statement. This does not teach or suggest adding to the FROM clause path expressions identified as candidates for an outer join.

The cited col. 11 discusses how if the collection of elements in "s.parts" is empty, no result is produced for that binding of "s". The plan that is generated for a query with outer quantifiers includes declarations to produce a singleton set if the set is bound to an empty set. The cited col. 11 discusses how the optimizer generates the plan for the query having a quantifier designated as outer and that the "for loops" are replaced with "for loops" marked with "outer". Nowhere does this cited col. 11 anywhere teach or suggest adding to the FROM statement path expressions identified as a candidate for a translation to an outer join. Instead, the cited col. 11 discusses how the "outer" quantifier is marked on the "for loops".

The Examiner cited the above discussed cols. 9-11 as teaching the claim requirement of grouping the ordered path expressions sequentially based upon a source-target dependency between ordered path expressions and based upon the identifications as a candidate for a translation to an outer join. (Office Action, pg. 4) Applicants traverse.

Applicants request the Examiner to identify which specific part of cols. 9-11 teaches the claim requirement of grouping the ordered path expressions sequentially based upon on a source-target dependency between ordered path expressions and based upon the identifications as a candidate for a translation to an outer join. The cited col. 9 mentions the use of an "outer" qualifier that applies to quantifier "s" and elements dependent on supplier "s". Although the cited col. 9 mentions the use of dependency to determine the quantifiers to which the qualifier "outer" applies, nowhere is there any teaching or suggestion of grouping ordered path expressions sequentially based upon a source-target dependency between ordered path expressions and upon the identification as a candidate for an outer join. Instead, the cited col. 9 looks at dependency to determine which quantifiers the "outer" qualifier.

The cited cols. 10-11 discuss how to implement outer quantifiers using nested loops to mark the "for loops" with outer. The Examiner has not cited any part of cols. 10-11 that teach or mention grouping ordered path expressions sequentially based upon a source-target dependency between ordered path expressions and upon the identification as a candidate for an outer join.

The Examiner cited col. 10, lines 15-42 of Cheng as teaching the claim requirements of creating a quantifier for each path expression, said quantifier comprising a variable representing a table in a relational database and replacing each grouped path expression with a corresponding quantifier and related table in a relational database. (Office Action, pg. 4) Applicants traverse.

The cited col. 10 mentions transforming a full join query into a union query of left outer join and right order joint. A column of constant 1 is added in the null producing operand of the right outer join to assure that the column value is null for the preserved tuples. By applying the "IS NULL" predicate after the right outer join, all matched rows are removed from the answer set. Nowhere does this cited col. 10 anywhere teach or suggest creating a quantifier for each path expression and replacing each grouped path expression with a corresponding quantifier. Instead, the cited col. 10 discusses how to add a constant to an operand of the right outer join.

Applicants submit that claims 1, 6, and 11 are patentable over the cited art because the cited Boyer and Cheng do not teach or suggest all the claim requirements. If the Examiner maintains the rejection, Applicants request that the Examiner show which specific elements of the cited columns teach or suggest the claim requirements.

Claims 2, 3, 5, 7, 8, 10, 12, 13, and 15 are patentable over the cited art because they depend from one of claims 1, 6, and 11. The following claims provide additional grounds of patentability over the cited art.

Claims 2, 7, and 12 depend from claims 1, 6, and 11 and additionally require performing optimization on the grouped quantifiers, said optimization identifying quantifiers which can be a candidate for a translation to an inner join; generating an outer join for each quantifier which remains after optimization a candidate for a translation to an outer join; and generating an inner join for each quantifier which remains after optimization a candidate for a translation to an inner join.

The Examiner cited the above discussed col. 10 of Cheng as teaching the additional requirements of these claims. (Office Action, pg. 5) Applicants traverse.

The cited col. 10 mentions transforming a full join query into a union query of left outer join and right order joint. A column of constant 1 or any other constant is added in the null-producing operand of the right outer join to assure that the column value is null for the preserved tuples. By applying the "IS NULL" predicate after the right outer join, all matched rows are removed from the answer set.

Nowhere does this cited col. 10 anywhere teach or suggest performing optimization on grouped quantifiers to identify quantifiers which can be a candidate for a translation to an inner join and generating an outer join for each quantifier which remains after optimization a candidate for a translation to an outer join. Further, nowhere does the cited col. 10 anywhere teach or suggest generating an inner join for each quantifier remaining after optimization a candidate for translation to an inner join. Instead, the cited col. 10 discusses how to add a constant to an operand of the right outer join so that all matched rows are removed from the answer set.

Accordingly, claims 2, 7, and 12 provide additional grounds of patentability over the cited art because the additional requirements of these claims are not taught or suggested in the cited art.

Claims 3, 8, and 13 depend from claims 2, 7, and 12 and additionally require that the optimization identifies a quantifier as a candidate for a translation to an inner join if a corresponding path expression is used in a FROM clause.

The Examiner cited col. 9, line 34 of Boyer as teaching the additional requirements of these claims. (Office Action, pg. 5) Applicants traverse.

The cited col. 9 is a from statement of "from Supplier s, (s.parts) p", where quantifier "s" is declared on suppliers and quantifier "p" is declared on the parts that "s" supplies Supplier, and the quantifier "p" are defined over the nested collection of parts embedded within each suppler. (Boyer, col. 9, lines 35-45).

Nowhere does the cited col. 9 teach or suggest that an optimization identify a quantifier as a candidate for a translation to an inner join if a corresponding path expression is used in a FROM clause. Instead, the cited col. 9 mentions quantifiers "s" and "p" declared on suppliers and parts, respectively.

Accordingly, claims 3, 8, and 13 provide additional grounds of patentability over the cited art because the additional requirements of these claims are not taught or suggested in the cited art.

2. <u>Claims 4, 9, and 14 are Patentable Over the Cited Art</u>

The Examiner rejected claims 4, 9, and 14 as obvious (35 U.S.C. §103(a)) over Boyer in view of Pirahesh (U.S. Patent No. 5,548,754). Applicants traverse.

These claims are patentable over the cited art because they depend from one of claims 2, 7, and 12, which are patentable over the cited art for the reasons discussed above. Moreover, the additional requirements of these claims provide further grounds of patentability over the cited art for the following reasons.

Claims 4, 9, and 14 depend from claims 2, 7, and 12 and additionally require that the optimization identifies a quantifier as a candidate for a translation to an inner join if a LIKE, IN, or BETWEEN operator exists in a WHERE clause containing a corresponding path expression.

The Examiner cited col. 9, lines 1-8 of Pirahesh as teaching the additional requirements of this claim. (Office Action, pg. 6) Applicants traverse.

The cited col. 9 discusses a query where an early-out join can be used on the subquery or inner query block even though the inner table column is referenced after the join in the subquery's select clause.

Nowhere does the cited col. 9 anywhere teach, suggest or mention that that the optimization identifies a quantifier as a candidate for a translation to an inner join if a LIKE, IN, or BETWEEN operator exists in a WHERE clause containing a corresponding path expression. If the Examiner maintains this rejection, Applicants request the Examiner to specifically show where this claim requirement is shown in the cited col. 9 if the rejection is maintained.

Accordingly, claims 4, 9, and 14 provide additional grounds of patentability over the cited art because the additional requirements of these claims are not taught or suggested in the cited art.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1-15 are patentable over the art of record. Applicants have not added any claims. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0460.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: August 2, 2006

By: /David Victor/______________

David W. Victor
Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984